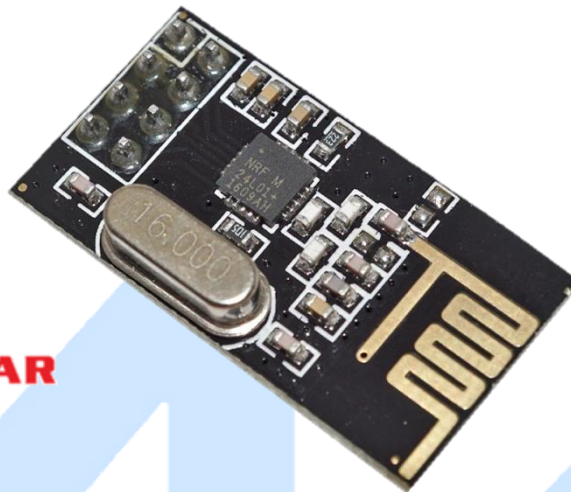


OKY3362

NRF24L01 MÓDULO TRANSECTOR INALÁMBRICO CON ANTENA DE 2.4 GHZ



Descripción:

NRF24L01 + Módulo transceptor inalámbrico con antena de 2,4 GHz. Velocidades de funcionamiento máximas de hasta 2 Mbps. Eficiencia de modulación GFSK. Capacidad anti interferente. Particularmente adecuado para aplicaciones de control industrial.

Características:

- Voltaje de alimentación: 1.9-3.6V (recomendado 3.3V)
- Potencia de salida máxima: + 20dBm
- Corriente del modo de emisión (pico): 115mA
- Corriente de modo de recepción (pico): 45 mA
- Corriente del modo de apagado: 4.2uA
- Tamaño de la placa: 33x14mm

 (¿Qué vamos a innovar hoy?)		AG Electrónica S.A.P.I. de C.V. República del Salvador N° 20 Segundo Piso Teléfono: (01)55 5130 - 7210	
ACOTACIÓN: N/A	http://www.agelectronica.com	ESCALA: N/A	REALIZO: SAA
			REV: JFRR
TOLERANCIA: N/A	NRF24L01 MÓDULO TRANSECTOR INALÁMBRICO CON ANTENA DE 2.4 GHZ		
TOLERANCIA: N/A	Fecha: 08/01/2021	No. Parte: OKY3362	

Conexión de la tarjeta



nRF24L01+ Pinout



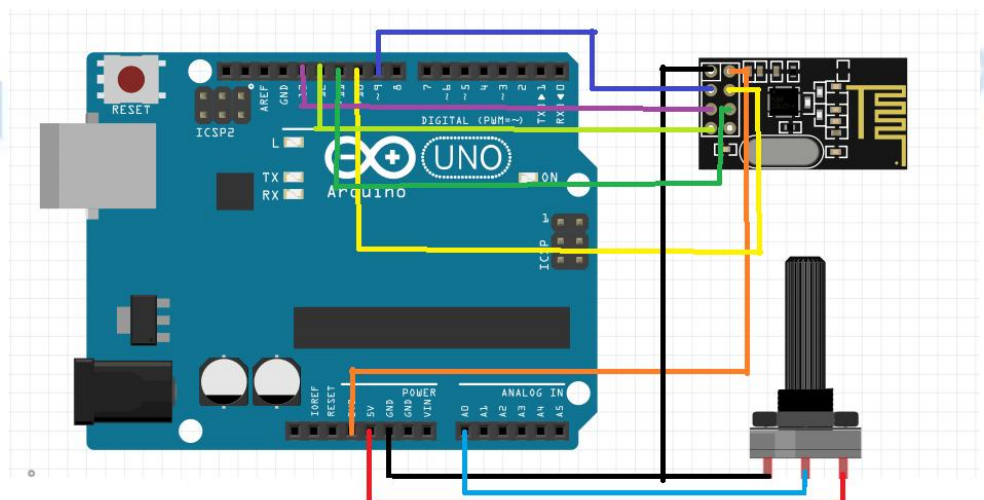
nRF24L01+ PA/LNA Pinout

Aplicación:

Envío de datos analógicos de lectura de un potenciómetro

Para esta aplicación sencilla estamos utilizando un entrada analógica de un potenciómetro para leer un valor determinado en el transmisor, el cual será enviado a un receptor. Para el módulo se utiliza una alimentación de 3.3V, los cuales pueden ser obtenidos del arduino(no recomendado por su baja corriente de salida) o una fuente externa(es mejor, pero la tierra debe estar conectada a la del arduino uno).

Conexión del Transmisor



Nota: Para el conexionado del receptor es de la misma manera, únicamente se omite el potenciómetro

Código para el transmisor

```
#include <SPI.h>
#include <nRF24L01.h>
#include <RF24.h>
//Declaremos los pines CE y el CSN
#define CE_PIN 9
#define CSN_PIN 10
//Variable con la dirección del canal por donde se va a transmitir byte direccion[5] ={'c','a','n','a','l'};
//creamos el objeto radio (NRF24L01)
RF24 radio(CE_PIN, CSN_PIN);
//vector con los datos a enviar float datos[3];
void setup()
{
    //inicializamos el NRF24L01
    radio.begin();
    //inicializamos el puerto serie
    Serial.begin(9600);
    //Abrimos un canal de escritura
    radio.openWritingPipe(direccion);
}
void loop()
{
    //cargamos los datos en la variable datos[]
    datos[0]=analogRead(0)* (5.0 / 1023.0);
    datos[1]=millis();
    datos[2]=3.14;
    //enviamos los datos
    bool ok = radio.write(datos, sizeof(datos));
    //reportamos por el puerto serial los datos enviados
    if(ok)
    {
        Serial.print("Datos enviados: ");
        Serial.print(datos[0]);
        Serial.print(" , ");
        Serial.print(datos[1]);
        Serial.print(" , ");
        Serial.println(datos[2]);
    }
    else
    {
        Serial.println("no se ha podido enviar");
    }
    delay(1000);
}

delay(1000);
}
```

Código para el Receptor

```
#include <SPI.h>
#include <nRF24L01.h>
#include <RF24.h>
//Declaremos los pines CE y el CSN
#define CE_PIN 9
#define CSN_PIN 10
//Variable con la dirección del canal que se va a leer
byte direccion[5] ={'c','a','n','a','l'};
//creamos el objeto radio (NRF24L01)
RF24 radio(CE_PIN, CSN_PIN);
//vector para los datos recibidos float datos[3];
void setup()
{
  //inicializamos el NRF24L01
  radio.begin();
  //inicializamos el puerto serie
  Serial.begin(9600);

  //Abrimos el canal de Lectura
  radio.openReadingPipe(1, direccion);

  //empezamos a escuchar por el canal
  radio.startListening();
}

void loop() {
  uint8_t numero_canal;
  //if ( radio.available(&numero_canal) )
  if ( radio.available() )
  {
    //Leemos los datos y los guardamos en la variable datos[]
    radio.read(datos,sizeof(datos));

    //reportamos por el puerto serial los datos recibidos
    Serial.print("Dato0= ");
    Serial.print(datos[0]);
    Serial.print(" V, ");
    Serial.print("Dato1= ");
    Serial.print(datos[1]);
    Serial.print(" ms, ");
    Serial.print("Dato2= ");
    Serial.println(datos[2]);
  }
  else
  {
    Serial.println("No hay datos de radio disponibles");
  }
  delay(1000);
}
```